

IN THE CLAIMS:

This listing of the claims replaces all prior versions and listings of the claims in this application.

The text of all pending claims (including any withdrawn claims) is set forth below. Canceled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~strikethrough~~. The status of each claim is indicated with one of (Original), (Currently amended), (Canceled), (Withdrawn), (Previously presented), (New), and (Not entered).

Please AMEND claims 19 and 29 in accordance with the following:

1. (Previously presented) An apparatus for reproducing audio video (AV) data using a markup document in an interactive mode selected by a user of the apparatus, comprising:
 - a buffer to buffer the markup document to enable the apparatus to reproduce the AV data in the interactive mode selected by the user; and
 - a buffer manager to manage the buffer to preload the markup document and output buffering state information of the buffer in response to a report signal, the buffering state information being used by the apparatus in reproducing the AV data in the interactive mode selected by the user.
2. (Previously presented) The apparatus of claim 1 further comprising a content decoder to interpret the markup document and output the report signal;
 - wherein the buffer manager informs the content decoder of the buffering state information of the buffer in response to the report signal.
3. (Previously presented) The apparatus of claim 2, wherein the content decoder generates the report signal using an application program interface (API).
4. (Previously presented) The apparatus of claim 3, wherein the API notifies the content decoder of whether preloading of the markup document has succeeded or failed, or whether the markup document is still being loaded.

5. (Previously presented) The apparatus of claim 4, wherein the API returns a value of 0 if the preloading of the markup document has succeeded, returns a value of 1 if the preloading of the markup document has failed, and returns a value of 2 if the markup document is still being loaded.

6. (Previously presented) The apparatus of claim 2, wherein the content decoder generates the report signal using an API comprising a file path and/or an attribute of the markup document as a parameter.

7. (Previously presented) The apparatus of claim 2, wherein the content decoder generates the report signal using an [obj].isCached(URL, resType) API, where URL is a parameter indicating a file path of the markup document, and resType is a parameter indicating an attribute of the markup document.

8. (Previously presented) The apparatus of claim 2, wherein the buffer manager informs the content decoder of a buffering state of the markup document using an API.

9. (Previously presented) The apparatus of claim 1, further comprising a content decoder to interpret the markup document;

wherein the buffer manager transfers the markup document from the buffer to the content decoder in response to a reproduce signal.

10. (Previously presented) The apparatus of claim 9, wherein the content decoder outputs a release signal to the buffer manager indicating that the markup document that was transferred from the buffer to the content decoder in response to the reproduce signal is not in use.

11. (Original) The apparatus of claim 10, wherein the content decoder outputs the release signal to the buffer manager in response to the markup document no longer being displayed in a screen of a display device.

12. (Previously presented) The apparatus of claim 1 further comprising a content decoder to interpret the markup document;
wherein the buffer manager deletes the markup document from the buffer in response to a discard signal output from the content decoder.

13. (Original) The apparatus of claim 12, wherein the content decoder generates the discard signal using a discard API.

14. (Previously presented) The apparatus of claim 2, wherein the content decoder generates the report signal using a progressNameOfFile API to determine a file name of the markup document currently being preloaded.

15. (Original) The apparatus of claim 2, wherein the content decoder generates the report signal using a progressLengthOfFile API to determine how much of the markup document currently being preloaded has been preloaded.

16. (Previously presented) The apparatus of claim 2, wherein the content decoder generates the report signal using a remainLengthOfFile API to determine how much of the markup document currently being preloaded is yet to be preloaded.

17. (Previously presented) The apparatus of claim 2, wherein the content decoder generates the report signal using a totalLoadingSize API to determine a total loading size of the markup document to be preloaded.

18. (Previously presented) The apparatus of claim 2, wherein the content decoder generates the report signal using a remainLoadingSize API to determine how much of a total loading size of the markup document has yet to be preloaded.

19. (Currently amended) An apparatus for controlling a buffer to buffer a markup document to reproduce audio video (AV) data in an interactive mode selected by a user of the apparatus, the apparatus comprising:

a physical element;

an electronic component;

a buffer manager to manage the buffer to preload the markup document to enable the apparatus to reproduce the AV data in the interactive mode selected by the user, and output information of the buffer comprising buffering information of the markup document;

wherein the buffering information is used by the apparatus in reproducing the AV data in the interactive mode selected by the user and comprises:

information indicating that preloading of the markup document has succeeded;

information indicating that the preloading of the markup document has failed; and

information indicating that the preloading of the markup document is still being conducted.

20. (Previously presented) The apparatus of claim 19, wherein the buffer manager outputs the information of the buffer using an application program interface (API).

21. (Previously presented) The apparatus of claim 19, wherein the information of the buffer further comprises information indicating whether a command to preload the markup document has been successfully received.

22. (Previously presented) The apparatus of claim 19, wherein the information of the buffer further comprises information indicating whether preloading of the markup document has been completed.

23. (Previously presented) An apparatus for recording and/or reproducing audio video (AV) data using a markup document in an interactive mode selected by a user of the apparatus before the apparatus reproduces any of the AV data, comprising:

an AV buffer to buffer the AV data;

an AV reproduction engine to decode the AV data;

an enhanced navigation (ENAV) buffer to preload the markup document before the apparatus reproduces any of the AV data to enable the apparatus to reproduce the AV data in the interactive mode selected by the user;

an ENAV engine to identify buffering state information of the markup document and decode the markup document, the buffering state information being used by the apparatus in reproducing the AV data in the interactive mode selected by the user; and
an I/O manager to obtain the markup document.

24. (Previously presented) The apparatus of claim 23, wherein the I/O manager uses a blocked I/O method to obtain the markup document from a data storage medium, and uses an unblocked I/O method to obtain the markup document from a network.

25. (Previously presented) The apparatus of claim 1, further comprising a reader to read a preload-list file before the reproducing of the AV data begins in the interactive mode selected by the user;

wherein the buffer manager manages the buffer to preload the markup document based on contents of the preload-list file before the reproducing of the AV data begins in the interactive mode selected by the user.

26. (Previously presented) The apparatus of claim 25, wherein the preload-list file contains information identifying at least one markup document that is to be preloaded into the buffer under control of the buffer manager before the reproducing of the AV data begins in the interactive mode selected by the user.

27. (Previously presented) The apparatus of claim 25, wherein the reader reads the preload-list file from an information storage medium.

28. (Previously presented) The apparatus of claim 1, wherein:
the interactive mode is a mode in which the AV data is displayed in a display window defined by the markup document;
the apparatus is selectively operable in the interactive mode in which the AV data is displayed in the display window defined by the markup document, and a non-interactive video mode in which the AV data is displayed in the same manner as AV data recorded on a standard DVD; and

the user of the apparatus selects between the interactive mode and the non-interactive video mode.

29. (Currently amended) The apparatus of claim ~~4~~ 19, wherein the ~~physical element~~ electronic component is a computer.

30. (Previously presented) The apparatus of claim 29, wherein the buffer manager is implemented by instructions performed by the computer.